Avaliação de Desempenho
Tópicos Especiais em Arquitetura de Computadores
Segundo Trabalho Prático
Pré-processamento de Dados

Observações gerais sobre este trabalho

1. Se for aluno de graduação, poderá fazer este trabalho individual, ou em duplas; se for aluno de pós-graduação terá que fazer o trabalho individual;
2. Você poderá programar em qualquer linguagem ou ambiente que você queira (Python, Shell script, R). Entretanto, a linguagem Python já tem muitas bibliotecas prontas (sci-kit learn, numpy e pandas, por exemplo).
3. Há muitos materiais disponíveis na web sobre pré-processamento de dados;
4. A primeira questão vale 4.0 pontos (0.8 para cada subquestão) e as três questões seguintes valem 2.0 pontos cada.

Considere a base de **mamografias** (Mamo), composta por 961 objetos e 6 atributos. Este exercício visa fazer a limpeza dos dados, para tratar os valores ausentes, além de aplicar métodos de discretização, transformação e redução de dados. É importante que a **limpeza dos dados seja sempre a primeira etapa**, para que os valores ausentes (ou ruídos) não interfiram nas próximas etapas de pré-processamento. Já a discretização, transformação e redução, podem ser intercaladas sem problemas.

Exercício 1. Limpeza dos dados - Valores Ausentes - Método da Imputação de Valores

No caso da base Mamo, há **162 valores ausentes** afetando 86 objetos diferentes. Uma das opções é eliminar esses objetos, reduzindo o tamanho da base, mas como a remoção pode causar perda de informação importante, nós não usaremos esta opção.

Aplique o **método de imputação de valores** para suprir a ausência dos valores dos atributos nos objetos. Considere que o atributo "***Severidade***" **determinará a classe dos objetos**, cujos valores são somente dois "benigno" ou "maligno", ou seja, só haverá duas classes para determinar os valores ausentes.

Exercício 1.1. Atributo BI-RADS

Especificação: o atributo "***BI-RADS***" está ausente em **dois** objetos (ID 21 e ID 209), sendo que cada um pertence a uma das classes, isto é, o ID 21 pertence a classe "**maligno**" e o ID 209 pertence a classe "**benigno**". Como o atributo é discreto, para determinar o valor que será atribuído aos objetos com valores ausentes, será utilizado **o valor de maior frequência** do atributo "BI-RADS" em cada classe. No caso específico da base Mamo, a maior frequência do atributo "BI-RADS" para a classe "maligno" é o

valor **5** (cuja frequência é igual a 427). A maior frequência do atributo “BI-RADS” para a classe “"benigno"” é o valor **4** (cuja frequência é igual a 306).

Entregável: implemente um programa que **calcule as frequências** do atributo “BI-RADS”, tanto para a classe "maligno" quanto para a classe "benigno", **descubra as linhas** que possuem valor ausente no atributo “BI-RADS” e gere um **novo *dataframe*** com os dados ausentes do atributo “BI-RADS” preenchidos.

Exercício 1.2. Atributo Idade

Especificação: o atributo “Idade” possui **5** objetos com valor ausente, sendo todos pertencentes à classe “maligno” (mas poderia não ser). Como o atributo é contínuo, é necessário calcular a **média dos valores** dos atributos dos objetos da mesma classe que, no caso específico da base Mamo, é 62,3. Como na definição do atributo “Idade” os valores são sempre inteiros, seria utilizado o valor **arredondado** de 62 para atribuir aos objetos com ID 444, 454, 684, 885 e 924.

Entregável: implemente um programa que calcule a média do atributo “Idade”, tanto para a classe "maligno" quanto para a classe "benigno", tomando o cuidado de arredondar para um valor inteiro, descubra as linhas que possuem valor ausente no atributo “Idade” e gere um novo dataframe com os dados ausentes do atributo “Idade” preenchidos.

Exercício 1.3. Atributo Forma

Especificação: no atributo “***Forma***” há **31** objetos sem valor, 19 da classe “benigno” e 12 da classe “maligno”. Na base Mamo, para a classe “benigno”, o valor mais frequente é “***redonda***”, presente em 186 objetos, ao passo que, para a classe “maligno”, o valor “***irregular***” está presente em 315 objetos, sendo o mais frequente. No caso específico da base Mamo, esses valores seriam imputados aos objetos que estão com valor ausente nesse atributo.

Entregável: implemente um programa que calcule as frequências do atributo “Forma”, tanto para a classe "maligno" quanto para a classe "benigno", descubra as linhas que possuem valor ausente no atributo “Forma” e gere um novo dataframe com os dados ausentes do atributo “Forma” preenchidos.

Exercício 1.4. Atributo Contorno

Especificação: o atributo “***Contorno***” possui **48** objetos sem valor, 37 da classe “benigno” e 11 da classe “maligno”. Para a classe “benigno”, o valor mais frequente é “**circunscrita**”, presente em 316 objetos; já para a classe “maligno”, o valor “**mal definida**” é o mais frequente e está presente em 191 objetos. No caso específico da base Mamo, esses valores seriam imputados aos objetos que estão com valor ausente nesse atributo.

Entregável: implemente um programa que calcule as frequências do atributo “Contorno”, tanto para a classe "maligno" quanto para a classe "benigno", descubra as linhas que possuem valor ausente no atributo “Contorno” e gere um novo dataframe com os dados ausentes do atributo “Contorno” preenchidos.

Exercício 1.5. Atributo Densidade

Especificação: o atributo “***Densidade***” possui **76** objetos sem valor, 54 da classe “benigno” e 22 da classe “maligno”. Para ambas as classes o valor mais frequente é “**baixa**” (mas poderia não ser), presente em 405 e 393 objetos das classes “benigno” e “maligno”, respectivamente. No caso específico da base Mamo, esse valor seria imputado a todos os objetos que estão com valor ausente nesse atributo.

Entregável: implemente um programa que calcule as frequências do atributo “Densidade”, tanto para a classe "maligno" quanto para a classe "benigno", descubra as linhas que possuem valor ausente no atributo “Densidade” e gere um novo dataframe com os dados ausentes do atributo “Densidade” preenchidos.

Exercício 2. Discretização - Método do encaixotamento de mesma largura

Especificação: muitos algoritmos em mineração de dados não trabalham com dados mistos, ou seja, bases de dados com atributos categóricos e contínuos. Na base Mamo, apenas o atributo “Idade” é contínuo e, portanto, ele será discretizado para que todos os atributos sejam categóricos, ampliando a gama de métodos de mineração que podem ser aplicados a essa base de dados. O atributo “**Idade**” será discretizado pelo método de **encaixotamento de mesma largura**, no qual os valores de cada caixa serão substituídos pela **média** (arredondada para um número inteiro) dos objetos contidos na caixa. Para determinar a largura das caixas precisamos encontrar o **menor** e o **maior** valor do atributo, que, no caso específico da base Mamo, são 18 e 96, respectivamente. Assumindo que a quantidade de caixas **será sempre de seis**, e dividindo o tamanho do intervalo entre o maior e o menor valor por seis, temos que cada caixa terá o intervalo de **13 unidades**. Para finalizar a discretização, basta substituir os valores de cada objeto, de cada caixa, pela média arredondada referente à caixa a qual ele pertence.

Entregável: implemente um programa que calcule o maior e o menor valor do atributo “Idade”, calcule a largura de cada caixa, gere a discretização baseado em seis caixas e gere um novo dataframe com os dados do atributo “Idade” ajustados para o valor da respectiva caixa.

Observação 1: considere a base de dados logo após ter passado pelo processo de limpeza de dados, ou seja, desconsidere a transformação e a redução.

Observação 2: vale a pena notar que, após a execução do programa, só haverão **seis valores** no atributo idade no novo dataframe.

Exercício 3. Transformação dos dados - Normalização Max-Min

Especificação: vamos aplicar a normalização somente no atributo "**Idade**". Considere que o intervalo de normalização será [0, 1]. Para aplicar essa normalização basta determinar os valores mínimo e máximo do atributo "Idade" e aplicar a equação de Normalização Max-Min em todas as linhas do dataframe.

Entregável: implemente um programa que aplique a equação de Normalização Max-Min considerando somente o atributo "Idade" e gere um novo dataframe contendo o valor da idade original e o valor normalizado entre 0 e 1.

Observação: considere a base de dados logo após ter passado pelo processo de limpeza de dados, ou seja, desconsidere a discretização e a redução.

Exercício 4. Redução dos dados - Amostragem Estratificada

Especificação: a redução dos dados tem como um dos objetivos a construção de bases de dados menores para aumento do desempenho computacional, ao mesmo tempo que tenta manter as características originais dos dados. Neste exemplo, será feita uma **amostragem estratificada** da base Mamo considerando o atributo "**Severidade**" como classe. Essa amostragem escolhe objetos aleatórios da base original para formar a nova base, sendo que a proporção entre o número de objetos de cada classe é mantida. Na base Mamo, há 516 objetos com "Severidade" igual a "benigno" e 445 com severidade igual a "maligno", ou seja, **53,7**% dos objetos com o valor "benigno" e **46,3%** com "maligno".

Entregável: implemente um programa que calcule o percentual de cada classe do atributo "Severidade", cujos valores são somente dois "benigno" ou "maligno", e aplique a amostragem estratificada, através da **seleção aleatória de 100 linhas** de dados, mas mantendo o mesmo percentual para cada classe, isto é, no caso da base mamo, em torno de **53,7%** para "benigno" e **46,3%** para "maligno". Esses valores podem ser aproximados.

Observação: considere a base de dados logo após ter passado pelo processo de limpeza de dados, ou seja, desconsidere a discretização e a transformação.